

AMENDMENTS TO THE CLAIMS

SAC
b1

1. (Currently Amended) A method comprising:
providing a first taken/not-taken prediction responsive to an address using a
saturating counter branch predictor;

providing (1) a second taken/not-taken prediction responsive to the address
resulting in a hit in a local branch history table, and (2) a hit/miss indication for the
address;

selecting for the address one of (1) the second prediction if the indication is a hit,
and (2) the first prediction if the indication is a miss; and

updating a ~~replacement~~ field for a matching entry in the local branch history
table only if the first prediction is incorrect, indicating that the entry is used to make a
prediction.

2. (Original) The method of claim 1 further comprising:

hashing the address prior to indexing at least one of the saturating counter
branch predictor and the local branch history table.

3. (Canceled)

4. (Previously Presented) The method of claim 1 further comprising:

fetching at least one instruction at the address, where if the instruction is a
branch, a determination as to whether the branch is taken or not-taken will not be
available until the instruction has progressed beyond a decode stage; and

decoding the at least one instruction, wherein at least one of the first and second
predictions is available when the at least one instruction is being decoded.

5. (Previously Presented) The method of claim 4 wherein the at least one
instruction is a branch, the method further comprising:

determining a target address of the branch; and

loading an instruction pointer generator with the target address if at least one of
the first and second predictions indicates that the branch is to be taken.

6. (Currently Amended) A processor comprising:

an instruction pointer (IP) generator capable of providing an address;

saturating counter branch prediction (SCBP) logic having an input coupled to the IP generator and capable of providing a first taken/not-taken prediction at an output responsive to the address;

local branch history prediction (LBHP) logic having an input coupled to the IP generator and capable of providing (1) a second taken/not-taken prediction at an output responsive to the address resulting in a hit, and (2) a hit/miss indication for the address, wherein the LBHP logic includes at least one local branch history table to provide a taken/not-taken history from a matching entry in said table in response to a said hit;

a multiplexer having an input coupled to the outputs of the SCBP and LBHP logic and a select input coupled to receive the hit/miss indication and in response provide (1) the second prediction if there is a hit and (2) the first prediction if there is a miss; and

entry replacement logic to update a replacement field for a said matching entry in the at least one table only if the first prediction is incorrect.

7. (Original) The processor of claim 6 further comprising:

address hash logic coupled between the IP generator and the inputs of the SCBP and LBHP logic to provide a plurality of index values to at least one of the SCBP and LBHP logic.

8. (Original) The processor of claim 6 wherein the SCBP logic includes a bimodal predictor.

9. (Original) The processor of claim 6 wherein the LBHP logic includes a plurality of local branch history tables each to provide a tag and a taken/not-taken history associated with the tag in response to a hit, compare logic coupled to each of the plurality of tables to determine the hit/miss indication, history multiplexer coupled to each of the plurality of tables to provide the history for the hit, and combinational logic coupled to an output of the history multiplexer to provide the second taken/not-taken prediction.

10. (Canceled)

11. (Original) The processor of claim 6 further comprising:
an instruction fetch stage of a pipeline; and
an instruction decode stage of the pipeline, and wherein the prediction at the output of the multiplexer is available when the address is being processed by an instruction decode stage of a pipeline.

12. (Original) The processor of claim 11 wherein the decode stage is capable of determining a target address of a branch instruction located at the address, the processor further comprising:
control logic coupled to load the IP generator with the branch target address if an output of the multiplexer indicates, for the address, that a branch is predicted to be taken.

b1

13. (Original) The processor of claim 6 wherein the address points to a cache line having a plurality of instructions.

14. (Currently Amended) An apparatus comprising:
means for providing an address of at least one instruction;
means for providing a first taken/not-taken branch prediction based upon the current state of a state machine and responsive to the address;
local branch history prediction (LBHP) logic having an input coupled to the address providing means and capable of providing (1) a second taken/not-taken prediction at an output responsive to the address resulting in a hit, and (2) a hit/miss indication for the address, wherein the LBHP logic includes at least one local branch history prediction table to provide a taken/not-taken history from a history field of a matching entry in said table in response to a said hit; and
a multiplexer having an input coupled to the outputs of the first prediction means and the LBHP logic and a select input coupled to receive the hit/miss indication and in response provide (1) the second prediction if there is a hit and (2) the first prediction if there is a miss; and
means for updating a replacement field for a matching entry in the at least one table said history field, with the outcome of an executed branch instruction that is

pointed to by said address, only if the first prediction is incorrect~~correct and the second prediction is not.~~

15. (Original) The apparatus of claim 14 further comprising:
means for encoding the address to provide a plurality of index values to at least one of the first prediction means and the LBHP logic.

16. (Original) The apparatus of claim 14 wherein the LBHP logic includes a plurality of local branch history prediction tables each to provide a tag and a taken/not-taken history associated with the tag in response to a hit, compare logic coupled to each of the plurality of tables to determine the hit/miss indication, history multiplexer coupled to each of the plurality of tables to provide the history for the hit, and combinational logic coupled to an output of the history multiplexer to provide the second taken/not-taken prediction.

17. (Canceled)

18. (Original) The apparatus of claim 14 further comprising:
means for fetching at least one instruction at the address; and
means for decoding the at least one instruction, wherein at least one of the first and second predictions is available when the at least one instruction is being decoded.

19. (Original) The apparatus of claim 18 wherein the at least one instruction is a branch, the apparatus further comprising:
means for determining a target address of the branch; and
means for fetching at least one instruction at the target address if at least one of the first and second predictions indicates that the branch is to be taken.

20. (New) A method comprising:
providing a first taken/not-taken prediction responsive to an address using a saturating counter branch predictor;

providing a second taken/not-taken prediction responsive to the address resulting in a hit in a local branch history table;

selecting for the address one of the first prediction and the second prediction depending on whether or not the address resulted in said hit; and

updating a replacement field, for an existing entry in the local branch history table that includes a history field containing a plurality of taken/not-taken outcomes of one or more previously executed branch instructions associated with said address, only if the first prediction is incorrect.

21. (New) A method comprising:

providing a first taken/not-taken prediction responsive to an address using a saturating counter branch predictor;

providing (1) a second taken/not-taken prediction responsive to the address resulting in a hit in a local branch history table, and (2) a hit/miss indication for the address, wherein the second prediction is based on a taken/not-taken history contained in a history field of an existing entry in said table;

selecting for the address one of (1) the second prediction if the indication is a hit, and (2) the first prediction if the indication is a miss; and

updating said history field based on an outcome of an executed branch instruction that is pointed to by said address, only if the first prediction is correct and the second prediction is not.
